CLAIMS

What is claimed is:

1. A pushback FIFO having an input and an output, the pushback FIFO allowing data values that have been unloaded from the pushback FIFO to be reloaded into the pushback FIFO at the beginning of a sequence of data values stored in the pushback FIFO if a determination is made that a data value should not have been unloaded from the pushback FIFO.

2. The pushback FIFO of claim 1, wherein said determination is made by logic external to the pushback FIFO and provided to the pushback FIFO.

3. The pushback FIFO of claim 1, wherein if a determination is made that the data value unloaded from the pushback FIFO should have been unloaded, the unloaded data value is not reloaded into the pushback FIFO.

4. The pushback FIFO of claim 3, wherein if a determination is made that the data value unloaded from the pushback FIFO should have been unloaded, the unloaded data value is marked as an invalid FIFO data value.

5. The pushback FIFO of claim 1, further comprising:

a standard FIFO having a plurality of storage locations for storing data values in a first-in-first-out fashion so that data values can be unloaded from the standard FIFO in a same sequence in which data values were loaded into the standard FIFO;

first logic, the first logic storing a copy of a data value unloaded from the standard FIFO;

second logic, the second logic outputting said data value unloaded from the standard FIFO and, if the data value output from the pushback FIFO should not have been output from the pushback FIFO, the second logic outputs the stored copy of the data value in a subsequent read cycle.

6. The pushback FIFO of claim 5, wherein said determination is made by logic external to the pushback FIFO and provided to the pushback FIFO.

7. The pushback FIFO of claim 5, wherein if a determination is made that the data value unloaded from the pushback FIFO should have been unloaded, the unloaded data value is not reloaded into the pushback FIFO.

8. The pushback FIFO of claim 7, wherein if a determination is made that the data value unloaded from the pushback FIFO should have been unloaded, the unloaded data value is marked as an invalid FIFO data value.

9. The pushback FIFO of claim 5, wherein the first logic includes a first multiplexer having at least first and second inputs and an output, the first input being connected to an output of the standard FIFO, the multiplexer receiving at least one control signal, said at least one control signal controlling whether a data value unloaded from the standard FIFO is to be output from the output of the first multiplexer.

10. The pushback FIFO of claim 9, wherein the first logic includes a storage element having an input connected to the output of the first multiplexer and an output connected to the second input of the first multiplexer, and wherein said at least one control signal controls whether a data value on the output of the storage element is to be output from the output of the first multiplexer.

11. The pushback FIFO of claim 10, further comprising second logic, the second logic comprising a second multiplexer, the second multiplexer having at least first and second inputs and an output, the first input of the second multiplexer being connected to the output of the standard FIFO and the second input of the multiplexer being connected to the output of the storage element, the second multiplexer being controlled by at least one control signal, and wherein said at least one control signal of the second multiplexer controls whether the data value unloaded from the standard FIFO will be output on the output of the second multiplexer or whether the data value on the output of the storage element will be output on the output of the second multiplexer, the output of the second multiplexer corresponding to the output of the pushback FIFO.

12. The pushback FIFO of claim 11, wherein if a determination is made that a data value unloaded from the pushback FIFO should not have been unloaded, on a next read cycle, the data value stored in the storage element will be output on the output of the second multiplexer and thereby output from the pushback FIFO.

13. The pushback FIFO of claim 11, wherein if a determination is made that a data value unloaded from the pushback FIFO should have been unloaded from the pushback FIFO, then on a next read cycle, a data value stored in the standard FIFO will be output on the output of the second multiplexer and thereby output from the pushback FIFO.

14. The pushback FIFO of claim 13, wherein said at least one control signal received by the first multiplexer determines whether a data value on the first input of the first multiplexer or a data value on the second input of the first multiplexer will be output on the output of the first multiplexer.

15. The pushback FIFO of claim 13, wherein said at least one control signal received by the second multiplexer determines whether a data value on the first input of the second multiplexer or a data value on the second input of the second multiplexer will be output on the output of the second multiplexer and thereby output from the pushback FIFO.

16. A method of performing a FIFO pushback operation, the method comprising the steps of:

unloading a first data value from a FIFO;

determining whether or not the unloaded first data value should have been unloaded from the FIFO;

if a determination is made that the unloaded first data value should not have been unloaded from the FIFO, reloading the first data value back into the FIFO such that the reloaded first data value occupies a first position in a sequence of data values stored in the FIFO.

17. The method of claim 16, wherein the reloading step includes:

storing a copy of the unloaded first data value in a storage element, the storage element being comprised by the FIFO; and

unloading the stored copy of the first data value from the storage element before unloading any other data value from the FIFO if a determination is made that the that the reloaded first data value should not have been unloaded from the FIFO

18. The method of claim 17, wherein if a determination is made that the first data value should have been unloaded from the FIFO, unloading a next data value from the FIFO.

19. The method of claim 18, further comprising the step of:

storing said next data value in the storage element.

20. The method of claim 19, further comprising the step of determining whether or not said next data value should have been unloaded from the FIFO, wherein if a determination is made that said next data value should not have been unloaded from memory, then on the next read cycle, unloading the stored next data value from the storage element, and wherein if a determination is made that said next data value should not have been unloaded from memory, then on said next read cycle, unloading a different next value from the FIFO.